/\*

|| @author Brett Hagman <bhagman@wiring.org.co>

|| @contribution Fotis Papadopoulos <fpapadopou@gmail.com>

|| @url http://wiring.org.co/

|| @url http://roguerobotics.com/

||

|| @description

|| | A Software Digital Square Wave Tone Generation Library

|| |

|| | Written by Brett Hagman

|| | http://www.roguerobotics.com/

|| | bhagman@roguerobotics.com, bhagman@wiring.org.co

|| |

|| | This is a Wiring Framework (Arduino) library to produce square-wave

|| | tones on an arbitrary pin.

|| |

|| | You can make multiple instances of the Tone object, to create tones on

|| | different pins.

|| |

|| | The number of tones that can be generated at the same time is limited

|| | by the number of hardware timers available on the hardware.

|| | (e.g. ATmega328 has 3 available timers, and the ATmega1280 has 6 timers)

|| |

|| | A simplified (single tone) version of this library has been included

|| | in the Wiring Framework since Wiring 0025 and in the Arduino distribution

|| | since Arduino 0018.

|| |

|| #

||

|| @license Please see the accompanying LICENSE.txt file for this project.

||

|| @name Software PWM Library

|| @type Library

|| @target Atmel AVR 8 Bit

||

|| @version 1.0.0

||

\*/

#ifndef \_Tone\_h

#define \_Tone\_h

#include <stdint.h>

/\*

|| Public Constants

\*/

#define NOTE\_B0 31

#define NOTE\_C1 33

#define NOTE\_CS1 35

#define NOTE\_D1 37

#define NOTE\_DS1 39

#define NOTE\_E1 41

#define NOTE\_F1 44

#define NOTE\_FS1 46

#define NOTE\_G1 49

#define NOTE\_GS1 52

#define NOTE\_A1 55

#define NOTE\_AS1 58

#define NOTE\_B1 62

#define NOTE\_C2 65

#define NOTE\_CS2 69

#define NOTE\_D2 73

#define NOTE\_DS2 78

#define NOTE\_E2 82

#define NOTE\_F2 87

#define NOTE\_FS2 93

#define NOTE\_G2 98

#define NOTE\_GS2 104

#define NOTE\_A2 110

#define NOTE\_AS2 117

#define NOTE\_B2 123

#define NOTE\_C3 131

#define NOTE\_CS3 139

#define NOTE\_D3 147

#define NOTE\_DS3 156

#define NOTE\_E3 165

#define NOTE\_F3 175

#define NOTE\_FS3 185

#define NOTE\_G3 196

#define NOTE\_GS3 208

#define NOTE\_A3 220

#define NOTE\_AS3 233

#define NOTE\_B3 247

#define NOTE\_C4 262

#define NOTE\_CS4 277

#define NOTE\_D4 294

#define NOTE\_DS4 311

#define NOTE\_E4 330

#define NOTE\_F4 349

#define NOTE\_FS4 370

#define NOTE\_G4 392

#define NOTE\_GS4 415

#define NOTE\_A4 440

#define NOTE\_AS4 466

#define NOTE\_B4 494

#define NOTE\_C5 523

#define NOTE\_CS5 554

#define NOTE\_D5 587

#define NOTE\_DS5 622

#define NOTE\_E5 659

#define NOTE\_F5 698

#define NOTE\_FS5 740

#define NOTE\_G5 784

#define NOTE\_GS5 831

#define NOTE\_A5 880

#define NOTE\_AS5 932

#define NOTE\_B5 988

#define NOTE\_C6 1047

#define NOTE\_CS6 1109

#define NOTE\_D6 1175

#define NOTE\_DS6 1245

#define NOTE\_E6 1319

#define NOTE\_F6 1397

#define NOTE\_FS6 1480

#define NOTE\_G6 1568

#define NOTE\_GS6 1661

#define NOTE\_A6 1760

#define NOTE\_AS6 1865

#define NOTE\_B6 1976

#define NOTE\_C7 2093

#define NOTE\_CS7 2217

#define NOTE\_D7 2349

#define NOTE\_DS7 2489

#define NOTE\_E7 2637

#define NOTE\_F7 2794

#define NOTE\_FS7 2960

#define NOTE\_G7 3136

#define NOTE\_GS7 3322

#define NOTE\_A7 3520

#define NOTE\_AS7 3729

#define NOTE\_B7 3951

#define NOTE\_C8 4186

#define NOTE\_CS8 4435

#define NOTE\_D8 4699

#define NOTE\_DS8 4978

/\*

|| Definitions

\*/

class Tone

{

public:

void begin(uint8\_t tonePin);

bool isPlaying();

void play(uint16\_t frequency, uint32\_t duration = 0);

void stop();

private:

static uint8\_t \_tone\_pin\_count;

uint8\_t \_pin;

int8\_t \_timer;

};

#endif